

Bakalářská práce



České
vysoké
učení technické
v Praze

F3

Fakulta elektrotechnická
Katedra počítačů

Aplikace pro podporu sdílení informací mezi čtenáři knih

Zuzana Štětinová

Vedoucí: Ing. Pavel Náplava, Ph.D.

Studijní program: Softwarové inženýrství a technologie

Leden 2020

I. OSOBNÍ A STUDIJNÍ ÚDAJE

Příjmení: **Štětinová**

Jméno: **Zuzana**

Osobní číslo: **456883**

Fakulta/ústav: **Fakulta elektrotechnická**

Zadávací katedra/ústav: **Katedra počítačů**

Studijní program: **Softwarové inženýrství a technologie**

II. ÚDAJE K BAKALÁŘSKÉ PRÁCI

Název bakalářské práce:

Aplikace pro podporu sdílení informací mezi čtenáři knih

Název bakalářské práce anglicky:

Application for sharing information among book readers

Pokyny pro vypracování:

Vytvořte aplikaci, která umožní čtenářům knih sdílet informace o obsahu jejich knihovny, recenzích přečtených knih, doporučeních pro ostatní čtenáře a vytváření „whishlistů“. Postupujte následujícím způsobem:

- 1) Analyzujte a definujte sadu klíčových požadavků, které vycházejí z reálných potřeb čtenářů, kteří chtějí mezi sebou sdílet informace o knihách.
- 2) Proveďte rešerši existujících řešení a vyhodnoťte jejich použitelnost z pohledu definovaných požadavků.
- 3) Na základě požadavků a nalezených řešení navrhnete a implementujete novou aplikaci, kterou bude možné pro specifikované účely využít. Alespoň v podobě alfa verze.
- 4) Funkčnost a použitelnost aplikace ověřte formou uživatelského testování na vybrané skupině čtenářů knih, ochotných sdílet informace.

Seznam doporučené literatury:

- [1] Michele BERTOLI. React Design Patterns and Best Practices. Packt Publishing 2017, ISBN 978-1786464538
- [2] Alex BANKS a Eve PORCELLO. Learning React. O'Reilly Media, Inc, USA 2017, ISBN 978-1491954621

Jméno a pracoviště vedoucí(ho) bakalářské práce:

Ing. Pavel Náplava, Ph.D., katedra ekonomiky, manažerství a humanitních věd FEL

Jméno a pracoviště druhé(ho) vedoucí(ho) nebo konzultanta(ky) bakalářské práce:

Datum zadání bakalářské práce: **14.02.2020**

Termín odevzdání bakalářské práce: **22.05.2020**

Platnost zadání bakalářské práce: **30.09.2021**

Ing. Pavel Náplava, Ph.D.
podpis vedoucí(ho) práce

podpis vedoucí(ho) ústavu/katedry

prof. Mgr. Petr Páta, Ph.D.
podpis děkana(ky)

III. PŘEVZETÍ ZADÁNÍ

Studentka bere na vědomí, že je povinna vypracovat bakalářskou práci samostatně, bez cizí pomoci, s výjimkou poskytnutých konzultací. Seznam použité literatury, jiných pramenů a jmen konzultantů je třeba uvést v bakalářské práci.

Datum převzetí zadání

Podpis studentky

Poděkování

Na tomto místě bych ráda poděkovala zejména svému vedoucímu Ing. Pavlu Náplavovi, Ph.D. za jeho odborné vedení, ochotu a celkovou vstřícnost při konzultacích a vypracování této bakalářské práce.

Dále bych také ráda poděkovala své rodině a přátelům za podporu a pomoc během studia.

Prohlášení

Čestně prohlašuji, že jsem bakalářskou práci na téma „Aplikace pro podporu sdílení informací mezi čtenáři knih“ vypracovala samostatně a s použitím uvedené literatury a pramenů v souladu s Metodickým pokynem o dodržování etických principů při přípravě vysokoškolských závěrečných prací.

.....
Zuzana Štětínová

V Praze, 22. 5. 2020

Abstrakt

Předmětem této práce je specifikace a realizace aplikace, která implementuje hlavní požadavky komunity knižních fanoušků získané z dotazníků a osobních konverzací. Aplikace se zaměřuje na snadnou rozšiřitelnost pro další vývojáře, kteří by si ji chtěli obohatit o další funkce nebo jazyky.

Klíčová slova: knihy, sdílení, čtenáři, webová aplikace, React, TypeScript, NodeJS

Vedoucí: Ing. Pavel Náplava, Ph.D.
Technická 2 ČVUT FEL
Praha 6, 166 27

Abstract

The subject of this thesis is the specification and creation of an application that implements key requirements obtained from the community of book fans via questionnaires and interviews. The application focuses on easy extensibility, to allow other developers to enhance e.g. the functionality or the language translations.

Keywords: books, sharing, readers, web application, React, TypeScript, NodeJS

Title translation: Application for sharing information among book readers

Obsah

1 Úvod	1	7.5 Závěry testování	36
2 Komunikace a sdílení informací	3	7.6 Shrnutí kapitoly	37
2.1 Informační kanály	3	8 Stav aplikace	39
2.2 Sociální skupiny	4	8.1 Současný stav	39
2.3 Shrnutí kapitoly	5	8.1.1 Nedostatky serverové části ..	39
3 Požadavky na aplikaci pro komunitu čtenářů	7	8.1.2 Nedostatky klientské části...	39
3.1 Funkční požadavky	7	8.2 Možná rozšíření	40
3.1.1 Požadavky čtenářské komunity	7	8.3 Shrnutí kapitoly	40
3.1.2 Požadavky osobních skupin...	9	9 Závěr	41
3.2 Nefunkční požadavky	10	Literatura	43
3.3 Shrnutí kapitoly	10		
4 Současná řešení	11		
4.1 Goodreads	11		
4.2 Databáze knih	12		
4.3 Shrnutí kapitoly	13		
5 Návrh aplikace	15		
5.1 Doménový model	15		
5.2 Případy užití a systémové požadavky	16		
5.3 Databázový model	21		
5.4 API	21		
5.5 Prototyp uživatelského rozhraní	22		
5.6 Shrnutí kapitoly	25		
6 Implementace a nasazení aplikace	27		
6.1 Architektura aplikace	27		
6.1.1 Databáze	27		
6.1.2 Serverová část	28		
6.1.3 Klientská část	28		
6.1.4 Použití TypeScriptu	29		
6.1.5 Sdílení kódu mezi klientskou a serverovou částí	29		
6.1.6 Zabezpečení aplikace	30		
6.2 Instalace a sestavení	30		
6.2.1 Potřebné technologie	31		
6.2.2 Popis instalace	31		
6.2.3 Popis spuštění aplikace	31		
6.3 Shrnutí kapitoly	32		
7 Testování aplikace	33		
7.1 Podmínky testování	33		
7.2 Cílová skupina	33		
7.3 Testovací scénáře	33		
7.4 Výsledky testování	34		

Obrázky

3.1 Výsledky dotazníku – současně používané aplikace a poptávka po nové	8
3.2 Výsledky dotazníku – Prioritizace požadavků	9
4.1 Hlavní stránka Goodreads	12
4.2 Hlavní stránka Databáze knih ..	13
5.1 Doménový model	15
5.2 Případy užití	20
5.3 Diagram databázového modelu .	21
5.4 Ukázka dotazů OpenAPI	22
5.5 Detail dotazu OpenAPI	23
5.6 Prototyp přihlášení a registrace .	24
5.7 Prototyp knihovny	25
5.8 Prototyp domovské stránky	26
6.1 Nasazení aplikace	27

Tabulky

4.1 Naplnění požadovaných funkcionalit stávajícími řešeními ..	14
---	----

Kapitola 1

Úvod

Knihy jsou i v dnešním světě velmi populární médium. Mnozí lidé čtou beletrii, učí se z učebnic, nebo čtou svým dětem pohádky na dobrou noc. S příchodem moderních technologií se objevily elektronické knihy (e-book) a k nim čtečky elektronických knih (e-reader). Vedle toho začaly vznikat mnohé aplikace s cílem čtenáře různými způsoby zaujmout. Tyto aplikace bývají převážně k dispozici on-line, nebo se dají stáhnout či koupit v obchodech s aplikacemi pro mobilní telefony. Různé aplikace se specializují na různé požadavky čtenářů. Příklady jsou internetová knihkupectví, aplikace fungující jako čtečky knih pro mobily a počítače, aplikace pro sdílení a čtení vlastní tvorby uživatelů, či soupisy čtenářských deníků, kde se sdílí mimo jiné obsah či myšlenka díla.

Samostatnou kategorií jsou aplikace, které aktivně pracují s knihovnou uživatele a umožňují ji kategorizovat. Ty velmi často podporují komunikaci mezi čtenáři, například pomocí číselného hodnocení a recenzování titulů, sdílení knihoven či přímé diskuze. V takovýchto aplikacích tak tvoří uživatelskou základnu převážně komunita vzájemně interagujících čtenářů.

Z mého pohledu však stále neexistuje žádná aplikace, která by dokázala efektivně zkombinovat nejčastější požadavky této čtenářské komunity. Současná řešení často čtenáře přehlcuje mnoha informacemi a funkcemi, které pro něj nejsou až tolik relevantní. Tato rozsáhlost tak brání jejich jednoduchému pochopení a použití. Na druhou stranu těmto aplikacím často chybí funkce, které by mnozí čtenáři uvítali. Příkladem je udržování zapůjčených knih nebo soukromé wishlisty. Se soukromím také souvisí to, že současná řešení tuto položku často úplně zanedbávají, všechna data se tak sdílí spíše do celé čtenářské komunity než pouze přátelům. Za zmínku také stojí, že u zahraničních aplikací často chybí příslušná jazyková lokalizace, což je další věc, která může některé čtenáře odradit.

Na základě zmíněných předpokladů jsem sesbírala informace k tomuto tématu pomocí hovorů se čtenáři a následně také internetového dotazníku. Zaměřila jsem se hlavně na českou a slovenskou čtenářskou komunitu, tedy komunitu považující za svůj mateřský jazyk češtinu nebo slovenštinu. Sebrané informace mi potvrdily, že mezi čtenáři existuje nespokojenost se stávajícími řešeními, a dokázaly i poptávku po novém jednodušším řešení. Na základě tohoto sběru informací jsem také stanovila hlavní požadavky pro tuto aplikaci

a míru jejich priority pro uživatele. Tyto požadavky jsou kombinací základních funkcí stávajících řešení a těch, které těmto řešením v současnosti často chybí.

Díky těmto závěrům jsem se rozhodla v rámci bakalářské práce zpracovat návrh a implementaci open-source aplikace splňující uvedené požadavky. Aplikace umožňuje uživateli správu vlastní knihovny, kde je možné si ke každé knize udržovat obecné a vlastní údaje, jako například hodnocení nebo poznámky. Uživatel si může zobrazit hodnocení od čtenářů, kteří jej zveřejní, či přátel. Další bližší interakce s ostatními uživateli je umožněna právě pomocí seznamu přátel. Uživatel si tak může zobrazit jejich soukromá hodnocení nebo třeba seznam knih, které si přejí. Další funkcionalitou je například udržování seznamu půjčených knih.

Tato práce si klade za cíl provést analýzu požadavků sledované komunity čtenářů, řešerši existujících řešení naplňujících některé tyto požadavky a na základě nalezených nedostatků návrh, implementaci alfa verze a uživatelské testování aplikace naplňující požadavky této komunity.

Práci jsem rozdělila na několik částí. V kapitole Komunikace a sdílení informací popisují teoretické základy komunikace a rozdělení sociálních skupin. Určila jsem zde cílovou skupinu aplikace a popsala komunikační principy, které aplikace využívá.

Následuje kapitola Požadavky na aplikaci pro komunitu čtenářů, ve které sepisují funkční a nefunkční požadavky v rámci cílových sociálních skupin.

V kapitole Současná řešení analyzuji existující řešení z části plnící definované požadavky. Poukazuji také na jejich nedostatky a možná vylepšení.

Tato vylepšení dále rozvádím návrhem aplikace v kapitole Návrh aplikace, kde jej postupně konkretizuji od obecných funkcionalit po architekturu a vzhled.

V kapitole Implementace a nasazení aplikace se zabývám konkrétní implementací aplikace, zvolenými technologiemi a samotnou instalací a spuštěním implementované alfa verze aplikace.

Kapitola Testování aplikace se zbývá ověřením použitelnosti navrženého řešení pomocí uživatelských testů aplikace.

Poslední kapitola Stav aplikace rozebírá současný stav aplikace a plány jejího rozšíření.

Kapitola 2

Komunikace a sdílení informací

Tato část práce se zabývá popisem mezilidské komunikace a rozbořem elektronických informačních kanálů a sociálních skupin. Zaměřuji se zde na volbu cílové skupiny aplikace a mapuji její komunikační prostředky.

Sdílení informací je důležitý proces, kterému se v běžném životě nevyhneme. Každá naše konverzace, přečtený článek nebo kniha, sledované video, či poslouchané rádio nám předává informace. Dle Cambridgeského slovníku [1] je sdílení informací jednání lidí, společností a organizací předávajících informace z jednoho na druhého, zejména elektronicky, nebo systém, který jim to umožňuje. Informace je tedy chápána jako obsah zprávy nebo sdělení.

Nadstavbou nad tímto sdílením je komunikace. Komunikace může mít vedle informační funkce i jiné motivace, jako vzájemnou domluvu, instruování, přesvědčování a podobné. Důležitým aspektem komunikace je také lidská potřeba bavit se, navazovat kontakty a sdružovat se.

Na rozdíl od pouhé výměny informací, komunikace zohledňuje i vzájemné působení a ovlivňování sdělení na základě kontextu. Vedle samotné informace tak zohledňuje osoby v konverzaci, záměr této konverzace a její účinek, ale samozřejmě také zvolený druh komunikace. Komunikaci tak můžeme dělit například na verbální a neverbální. Další možné dělení je na intrapersonální komunikaci (vnitřní monolog), interpersonální komunikaci (mezi více lidmi) a masovou komunikaci (jednosměrný proud informací od zdroje k mnoha příjemcům). V této práci se v rámci vybraného tématu dále budu soustředit na interpersonální komunikaci.

Vliv na komunikaci má ve velké míře také zvolený způsob předání informace. Tyto způsoby jsou označovány jako informační kanály a v následující sekci budou dále kategorizovány.

2.1 Informační kanály

Informační kanály mají mnoho různých parametrů, podle kterých se dají rozdělit. Jedno z nejpoužívanějších dělení je na formální a neformální komunikaci. Neformální komunikace má značný vliv na formování vztahu mezi komunikujícími, je daleko subjektivnější.

Informační kanály mohou být buďto tradiční, tedy probíhající osobně, jako například rozhovor, nebo elektronické. Elektronické kanály mají určité výhody,

všichni členové navzájem často ani neznají a skupinu tak spojuje společný zájem nebo cíl (například stát). Vazby mezi členy skupiny jsou děleny na primární, kde je hlavní motivací samotný vztah mezi členy, a sekundární, kde je hlavní motivací cíl nebo myšlenka skupiny. Související je také dělení na formální a neformální sociální skupiny.

Sociální skupiny vznikají jak osobním kontaktem, tak v rámci webových informačních kanálů, zejména on-line fór a sociálních sítí. Častou motivací vzniku neformálních skupin je společný zájem členů skupiny. Skupiny tak vznikají kupříkladu kolem různých škol či sportů, vaření, nebo podobného typu humoru. Jedním z těchto zájmů je i četba, kolem které vznikají sociální skupiny čtenářů.

Knihy jsou médiem těšícím se stále vysoké popularitě. Vyskytují se v mnoha formách. Vedle běžných papírově vydávaných knih se prodávají i e-knihy a audioknihy. Komunita čtenářů se spojuje napříč všemi těmito formami. Společným zájmem je zde obsah a myšlenka díla převládá nad formou. Čtenáře často dělí jazyková bariéra, čímž vznikají různé komunity podle jazyka vydání knih. Já se dále soustředím na komunity českých a slovenských čtenářů. Menší čtenářské skupiny se sdružují nad žánry nebo nad autory.

V těchto skupinách funguje komunikace na základě sdílení informací a názorů. Uživatelé sdílí své přečtené knihy, knihy, které právě čtou, či chtějí přečíst. Často sdílí i knihy, které vlastní nebo by si přáli vlastnit. Oblíbené je také sdílení hodnocení titulů.

Knihy jsou často společným zájmem v již existujících osobnějších sociálních skupinách, s čímž se můžeme setkat například mezi členy rodiny či kamarády.

Čtenáři sdružující se do různých komunit často ke komunikaci mezi sebou používají různé webové a mobilní aplikace. Tyto aplikace podporují většinou různé funkcionality pro sdílení mezi čtenářskou komunitou, dle mého názoru je však problém absence jedné aplikace, která by fungovala zároveň pro čtenářskou komunitu a osobní komunity čtenáře.

2.3 Shrnutí kapitoly

Komunikace není jen výměna informací, ale hlavně prostředek lidské socializace a vzájemné domluvy. Veškeré naše chování je komunikace, a tak ani nemůžeme nekomunikovat. Pro účely komunikace slouží komunikační kanály – tradiční a elektronické. Elektronické kanály mohou být různých typů a sloužit k rozličným typům komunikace. Nejdůležitějšími internetovými textovými informačními kanály jsou emaily, chatovací služby, komunikační platformy a sociální sítě. Dalšími důležitými kanály jsou aplikace, které mají komunikační funkcionality jako sekundární. Opakovaná komunikace funguje zejména v rámci sociálních skupin. Ty mohou mít různé stupně formality, velikosti a motivace. Sociální skupiny vznikají kolem společných zájmů osobně i v rámci on-line aplikací.

Na základě provedeného průzkumu jsem si upřesnila cílovou sociální skupinu a konkretizovala komunikační kanál, který bude použitý v navrhované aplikaci.

Základem aplikace je propojení hlavních požadavků pro čtenářskou komunitu a požadavků využitelných pro malé osobní komunity čtenáře. Možnost komunikace je jedna z hlavních funkcionalit aplikace, ne však jediná.

Kapitola 3

Požadavky na aplikaci pro komunitu čtenářů

Tato kapitola se zabývá sběrem a rozбором požadavků získaných od cílových skupin určených v předchozí kapitole. Sepsané požadavky jsou zohledňovány dále v návrhu a implementaci navrhované aplikace.

Sběr požadavků pro vypracovávanou aplikaci jsem zahájila tím, že jsem sepsala vlastní požadavky, které jsem diskutovala s mými známými. Poté jsem analyzovala stávající řešení aplikací pro komunikaci mezi čtenáři, viz kapitolu Současná řešení. Pokračovala jsem hovory s různými čtenáři, kde jsem zjišťovala, zda používají takovou aplikaci, a co jim na současných aplikacích chybí. Posledním bodem byl krátký dotazník mapující poptávku po nové aplikaci a prioritizující sesbírané požadavky. Zkrácené výsledky tohoto dotazníku jsou zobrazeny na obrázcích 3.1 a 3.2. Na 3.2 odpovídá rostoucí priorita rostoucímu číselnému hodnocení.

Rozbor výstupů z celého průzkumu je obsažen v následujících podkapitolách.

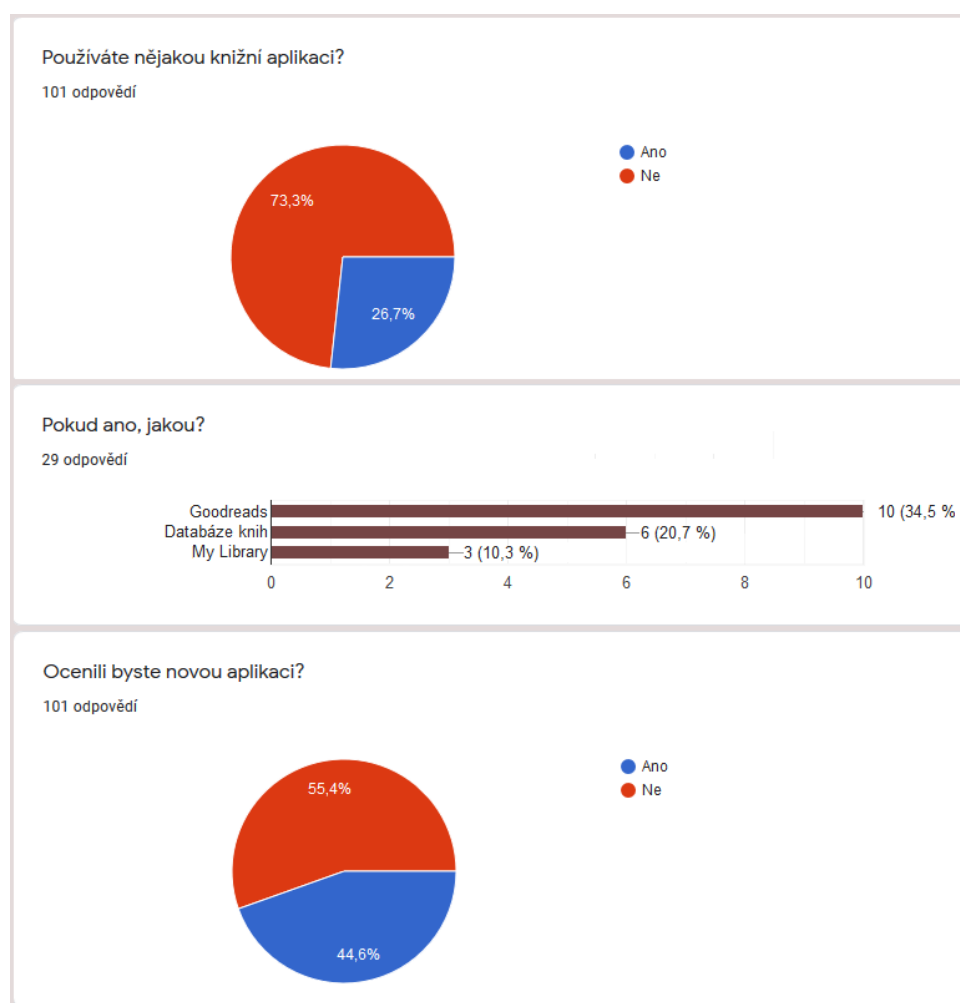
3.1 Funkční požadavky

Funkční požadavky jsou rozděleny do dvou kategorií, podle toho, v rámci jaké sociální skupiny je uživatel vyžaduje. Jmenovitě jde o čtenářskou komunitu a osobní skupinu uživatele, tedy například jeho rodina či přátelé.

3.1.1 Požadavky čtenářské komunity

Jako hlavní požadavky související s komunikací v rámci čtenářské komunity jsem identifikovala následující:

1. *Seznam knih:* Uživatelé požadují možnost udržování, prohlížení a třídění seznamu knih, například pomocí vlastních štítků.
2. *Údaje o každé knize:* Uživatelé požadují možnost zadat ke každé knize vlastní údaje. Povinné údaje by se měly udržet na minimu, aby uživatelé jejich zadávání neobtěžovalo, volitelné údaje naopak mohou být rozsáhlejší. Povinnými údaji určenými uživateli jsou:

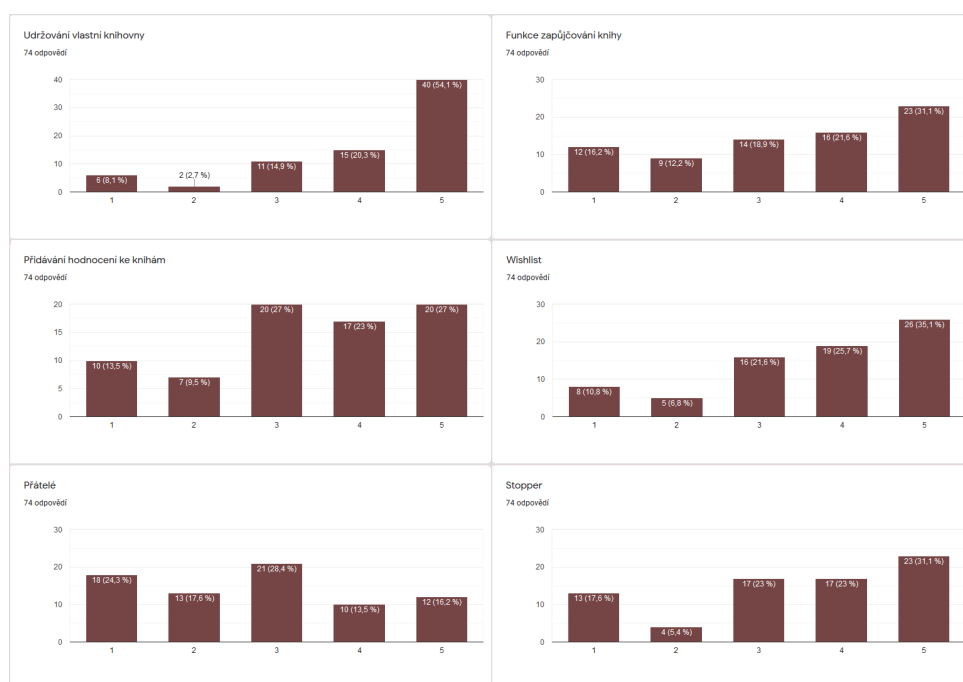


Obrázek 3.1: Výsledky dotazníku – současně používané aplikace a poptávka po nové

- Jméno knihy
- Autor / autoři

Volitelnými údaji je myšleno:

- Forma knihy (klasická, elektronická, audio)
- Nakladatelství
- Rok vydání
- ISBN
- Žánr
- Děj
- Datum dočtení
- Vlastní hodnocení – numerické (hvězdy) a slovní
- Vlastní poznámky



Obrázek 3.2: Výsledky dotazníku – Prioritizace požadavků

3. *Zobrazení recenzí:* uživatelé požadují možnost vyhledávat knihy pomocí jejich jména nebo autora a zobrazovat si recenze ostatních uživatelů.

3.1.2 Požadavky osobních skupin

Další požadavky souvisejí s osobními skupinami uživatele. Slouží převážně uživatelům, kteří se osobně znají. Zásadními požadavky tohoto typu jsou:

1. *Přidávání uživatelů mezi přátele:* Uživatelé požadují možnost přidat si jiné uživatele, které osobně znají, mezi přátele, například pomocí emailu.
2. *Zobrazení knih přátel:* Uživatelé požadují možnost si zobrazit knihy svých přátel a jejich hodnocení.
3. *Správa seznamů přání:* Uživatelé požadují možnost přidání knihy do seznamu přání (wishlistu). Uživatelé chtějí mít možnost zobrazení wishlistů svých přátel. Další požadovanou funkcionalitou je vybrání knihy z wishlistu přátel a její označení. Toto označení vidí ostatní přátelé, aby se zamezilo tomu, že různí přátelé koupí stejnou knihu. Majitel wishlistu vidí pouze "stopku" – označení, že někdo mu zakoupil nějakou knihu – bez dalších informací.
4. *Možnost zapůjčení knihy:* Uživatelé požadují možnost označení knihy jako zapůjčené. U zápůjčky je možné zaznamenat komu (uživatel aplikace nebo jen jméno) a kdy byla kniha zapůjčena.
5. *Nastavení soukromí:* Uživatelé požadují možnost nastavení soukromí profilu a recenzí.

Z průzkumu vyplynuly či byly přímo řečeny tyto nefunkční požadavky:

1. *Přehlednost a jednoduché ovládání:* Uživatelé požadují jednoduchou intuitivní orientaci v aplikaci, všechny části aplikace musí být jednoduše dosažitelné a pochopitelné.
2. *Ovládání na různých zařízeních:* Uživateli je požadováno, aby aplikace fungovala na počítačích, mobilech a tabletech různých velikostí.
3. *Podpora prohlížečů:* Uživateli je požadováno, aby aplikace fungovala na současných verzích prohlížečů Chrome, Edge, Safari a Firefox.
4. *Open-source:* Je požadováno, aby aplikace byla open-source (není požadována podpora aplikace, pouze nasazení na testovací server).
5. *Rozšiřitelnost:* Z principu open-source je požadavkem rozšiřitelnost pro budoucí vývojáře a další funkcionality.
6. *Bezpečnost:* Aplikace musí bezpečně zacházet s uživatelskými údaji. Je požadováno přihlašování pomocí služby Google a zabezpečená databáze uživatelských údajů.

Funkční požadavky na aplikaci, implementovanou v této bakalářské práci, se dají rozdělit na požadavky v rámci čtenářské komunity a požadavky v rámci okruhu známých. V komunitě je požadováno, aby aplikace udržovala knihy uživatele, umožňovala o nich zadávat údaje a přidávat k nim recenze. V okruhu známých je požadován seznam přátel, zobrazení jejich knih, správa seznamů přání a výpůjček knih. Uživatel také musí mít možnost nastavení soukromí pro oddělení těchto dvou okruhů.

Aplikace dále musí splňovat nefunkční požadavky jako jednoduchost, přehlednost a rozšiřitelnost v rámci open-source principu. Aplikace musí fungovat na současných běžných prohlížečích a umožňovat bezpečné přihlašování pomocí služby Google.

Kapitola 4

Současná řešení

V této kapitole se zaměřuji na aplikace s funkcionalitou popsanou v kapitole Požadavky na aplikaci pro komunitu čtenářů, především na ty, které jsou používané českou komunitou čtenářů. Výběr konkrétních aplikací je ovlivněn především osobními hovory, dotazníkem a samotnou funkcionální blízkostí s aplikací, které se tato práce dále věnuje.

Cílem této kapitoly je především zhodnotit smysluplnost navrhované aplikace a zjistit, zda již neexistuje adekvátní řešení.

4.1 Goodreads

Webová aplikace Goodreads [3], spravovaná společností Goodreads Inc., je velmi populární čtenářská aplikace, která má dle svých statistik [4] kolem 90 milionů členů. Toto členství vzniká pouze registrací, aplikace není placená. Ukázka aplikace je zobrazena na obrázku 4.1.

Databáze knih v aplikaci Goodreads je vytvářena uživateli a spravována administrátory. Kterýkoli uživatel aplikace v ní může vyhledávat pomocí zadání textu. Každá kniha zde obsahuje jméno, autora, krátké shrnutí děje a další informace o vydání, žánru a podobně. Navíc má kniha rating mezi jednou a pěti hvězdičkami, který přidávají sami členové. Ke knize může jakýkoli člen přidat recenzi, kterou si mohou všichni uživatelé stránky přečíst. Pro členy je možné na takovéto recenze odpovídat a hodnotit je. Členové mají také možnosti se například zeptat autora na otázky ohledně této knihy, vytvořit o ní diskuzi, či z ní citovat.

Další základní členská funkce je udržování vlastní knihovny. Člen si může sledovat, které knihy již přečetl, které právě čte a které si teprve chce přečíst, případně si založit další kategorizovací štítek. Knihy se zadávají pomocí vyhledávání. Knihu, která není ve vyhledávání, lze přidat do celkové databáze, čímž často vznikají duplikáty, zvláště jazykové.

Funguje zde také síť přátel, u kterých tak lze zjistit například co zrovna čtou nebo informace o jejich profilu a aktivitě. Aplikace mimo jiné umísťuje recenze přátel do vlastního oddělení před recenze ostatních.

Goodreads je velmi rozsáhlá mezinárodní komunitní aplikace, která nabízí velké množství obsahu, například výzvy a soutěže. Spolupracuje s on-line obchody jako Amazon a další, pod každou známější knihou jsou odkazy na



Obrázek 4.1: Hlavní stránka Goodreads

místa, kde se dá zakoupit. Kvůli množství dat je schopná svým členům na základě jejich zadaných knih doporučovat další, které by se jim mohly líbit.

Z hlediska častějšího užívání však vidím jako problém této aplikace příliš velký rozsah. Stránka toho nabízí tolik, až se stává nepřehlednou. Má to jistě svůj účel pro uživatele, kteří chtějí na této stránce strávit dlouhou dobu, ale může to působit nepříjemně na uživatele, který chce používat pouze základní funkce aplikace. Stránka se stává ještě nepřehlednější na mobilních zařízeních.

4.2 Databáze knih

Databáze knih [5] je česky lokalizovaná aplikace provozovaná společností Databáze knih s. r. o., která cílí na lokální čtenářskou komunitu. Podle svých slov [6] je nejnavštěvovanějším českým knižním webem, v současnosti ji dle jejích statistik navštěvuje průměrně 75 000 uživatelů denně. Aplikace je financována pomocí reklam provozovaných Seznamem. Její úvodní stránka je zobrazena na obrázku 4.2.

Tato aplikace spravuje svou knižní databázi pomocí administrátorů, kteří tyto knihy do aplikace přidávají. Pokud zde kniha není zaznamenaná, neexistuje pro uživatele možnost ji přidat. Vyhledávání v této aplikaci je možné pomocí textu nebo žánru, případně jiných předpřipravených filtrů. Knihy zde obsahují vždy základní údaje, jako jsou název, autor, žánr, originální název a stručný děj. Uživatelé mohou knihu hodnotit pomocí hvězdiček a do komentářové sekce pod knihou přidávat svoje názory a myšlenky ohledně této knihy. Ke knihám si uživatel může zobrazit různé zajímavosti, jejich vydání, případně se pokusit najít tuto knihu na knižním bazaru této stránky.

Knižní databáze nabízí u každé knihy seznam dalších knih autora, podobných knih, nebo například statistiku, kolik registrovaných uživatelů má tuto knihu v přečtených, právě ji čte a jiné. Aplikace také spolupracuje s internetovým obchodem Alza pomocí nabídky možností koupit knihu či koupit



Obrázek 4.2: Hlavní stránka Databáze knih

e-knihu.

I zde si uživatel může po přihlášení udržovat přehled své knihovny. U každé knihy má možnost si zvolit přidání do seznamu knih, kde si vybere jednu či více z možných kategorií a knihu si tak přidá do svého účtu. Základními kategoriemi jsou zde například přečtené knihy, právě čtené knihy, doporučované knihy a podobně. Uživatel ale není těmito kategoriemi omezován a má možnost vytvořit si vlastní.

Pro komunikaci s ostatními členy komunity se musí uživatel také přihlásit. Přihlášený uživatel má založený profil, kde o sobě může zveřejnit různé informace. Knižní seznamy uživatele jsou veřejně dostupné. Také může posílat přímé zprávy ostatním uživatelům, nebo si je přidat do svého seznamu oblíbených, a tak je sledovat. V aplikaci se nachází i možnost diskuzí nad různými tématy týkajícími se knih.

Databáze knih nabízí, jak již bylo zmíněno, také bazar knih, do kterého přidávají knihy sami uživatelé. Další zajímavou funkcí je zobrazení žebříčků nejlepších knih, autorů či nejaktivnějších uživatelů a mnohé další. Databáze knih také poskytuje doporučení, ale to je založeno primárně na vyhledávání uživatele.

Z mého pohledu je tato aplikace velmi užitečná, podobně jako Česko-Slovenská filmová databáze [7], nicméně mi přijde, že se zaměřuje pouze na celkovou čtenářskou komunitu a zanedbává osobnější kontakt jejích uživatelů. Aplikace také neumožňuje nastavení soukromí profilu.

4.3 Shrnutí kapitoly

Z aplikací splňujících do jisté míry výše zmíněné požadavky jsem vybrala Goodreads a Databázi knih právě kvůli tomu, že splňovaly nadpoloviční většinu požadavků. Dalším důvodem bylo, že ve sledované komunitě (čeští

Požadavek	Goodreads	Databáze knih
Seznam knih	Ano	Ano
Údaje o každé knize	Ano	Ano
Zobrazení recenzí	Ano	Ano
Přidávání uživatelů mezi přátele	Ano	Ne
Zobrazení knih přátel (ostatních uživatelů)	Ano	Ano
Správa seznamu přání	Ne	Částečně
Možnost zapůjčení knihy	Ne	Ne
Nastavení soukromí	Ne	Ne
Přehlednost a jednoduché ovládání – subjektivně	Částečně	Ano
Ovládání na všech zařízeních	Ano, i aplikace	Ano
Open-source	Ne	Ne
Přihlašování pomocí Google	Ano	Ano

Tabulka 4.1: Naplnění požadovaných funkcionalit stávajícími řešeními

a slovenští čtenáři) v dotazníku vyšly jako nejpoužívanější. Goodreads má za sebou větší základnu a podporu a má více funkcionalit, na druhou stranu databáze knih je dle mého názoru přehlednější a má českou lokalizaci.

Tabulka 4.1 v této kapitole zobrazuje, jaké ze sesbíraných požadavků splňují Goodreads a Databáze knih. Můžeme na ní pozorovat například to, že ani jedna aplikace nepodporuje možnost zapůjčení knih nebo nastavení soukromí. Aplikace také splňují různé požadavky. Zatímco Databáze knih částečně podporuje seznamy přání, nepodporuje udržování seznamu přátel, u Goodreads je tomu přesně naopak. Navíc obě aplikace jsou komerční, takže nejsou Open-source.

Z tabulky vyplývá, že navrhovaná aplikace rozšiřuje funkcionality oproti oběma zkoumaným řešením.

Kapitola 5

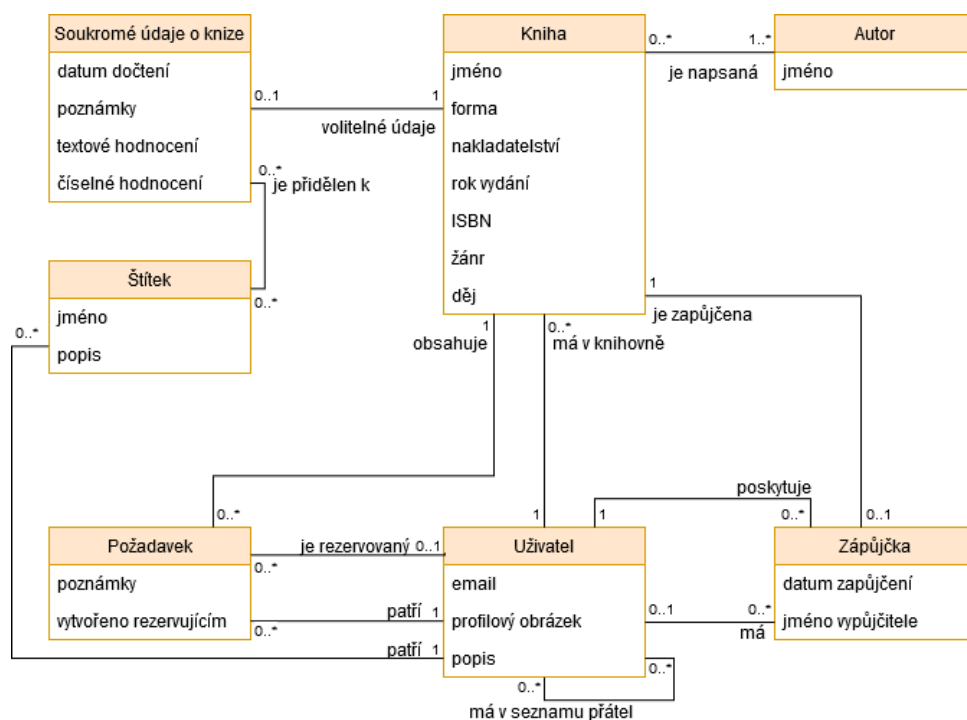
Návrh aplikace

V této kapitole popisují návrh aplikace, který vychází z uvedených sesbíraných požadavků v kapitole Funkční požadavky a částečně se inspiruje existujícími řešeními z kapitoly Současná řešení.

Aplikace je implementována jako vícevrstvá webová aplikace, skládá se z klientské části (front end), serverové části (back end) a databáze.

Návrh aplikace obsahuje doménový model, případy užití, související katalog systémových požadavků a databázový model. Následuje popis API, které poskytuje serverová část aplikace. Poslední částí návrhu je prototyp aplikace.

5.1 Doménový model



Obrázek 5.1: Doménový model

Návrh doménového modelu na obrázku 5.1 specifikuje data uživatelů a vztahy mezi nimi.

Hlavní entitou je *Kniha*, obsahující všechny důležité obecné informace o každé knize v knihovně uživatele. Tato entita je využívána také v seznamech zapůjčených knih a wishlistů.

Každá kniha má svého *Autora*, či více autorů.

Ke knize mohou být v rámci knihovny uživatele přidány *Soukromé údaje o knize*, které obsahují subjektivní údaje a názory čtenáře.

Uživatelé jsou registrovaní uživatelé aplikace, kteří vlastní seznamy knih, seznamy zápůjček a wishlisty.

Uživatelé si mohou vytvářet *Štítky* pro kategorizaci a filtrování jejich knihovny.

Wishlisty uživatelů jsou formovány pomocí *Požadavků*, které patří vybranému uživateli, ať již jsou vytvořeny jím, nebo tím, kdo je rezervuje.

Seznamy knih zapůjčených z knihovny jsou realizovány *Zápůjčkami*, které jsou poskytovány daným uživatelem.

5.2 Případy užití a systémové požadavky

Pro aplikaci jsem identifikovala případy užití a zavedla je do diagramu na obrázku 5.2. K těmto případům užití jsem sepsala katalog systémových požadavků, které jednotlivým případům užití odpovídají.

1. Přihlašování a uživatelský profil

- a. *Registrace*: Systém umožňuje uživateli registrovat se. Registrace je možná pomocí účtu Google, nebo pomocí emailu a zvoleného hesla, přičemž k tomu bude nutné potvrzení emailu. Při registraci může uživatel zadat své jméno, přidat obrázek, případně poznámku o sobě, aby ho přátelé lépe rozeznali. Již při registraci lze nastavit soukromí profilu.
- b. *Přihlášení*: Systém umožňuje registrovanému uživateli se přihlásit. Přihlášení si může uživatel zvolit podle registrace pomocí Google nebo emailu a hesla.
- c. *Odhlášení*: Systém umožňuje přihlášenému uživateli se kdykoli odhlásit.
- d. *Zobrazení profilu*: Systém umožňuje přihlášenému uživateli zobrazit si svůj profil.
- e. *Nastavení profilu*: Systém umožňuje v profilu změnit jméno, obrázek nebo poznámku. V profilu lze také nastavit změnu soukromí profilu.
- f. *Zrušení účtu*: Systém umožňuje vytvořený účet a všechna data s ním spjatá odstranit. Odstranění musí být potvrzeno, aby nedošlo ke smazání omylem.

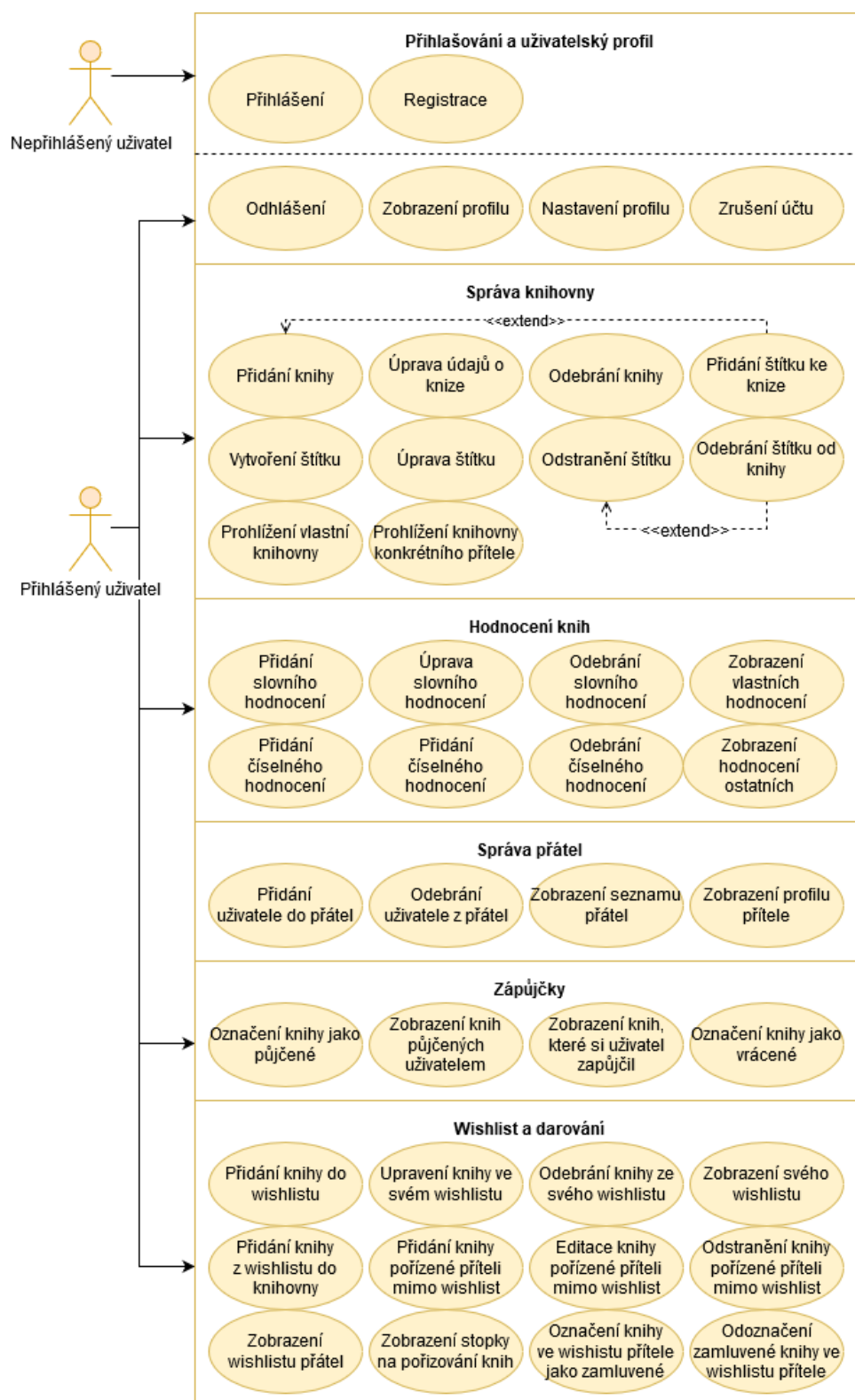
2. Správa knihovny

- a. *Přidání knihy:* Systém umožňuje uživateli přidat si knihu do vlastní knihovny. Kniha je identifikovaná pomocí jména a autora (či autorů). K těmto povinným údajům je možné přidat další volitelné údaje. Těmi jsou forma knihy (klasická, elektronická, audio), nakladatelství, rok vydání, ISBN, žánr, děj, datum dočtení či vlastní poznámky. Systém umožňuje současně přidat ke knize štítky.
- b. *Úprava údajů o knize:* Systém umožňuje přihlášenému uživateli upravit všechny volitelné údaje o jakékoli knize v jeho knihovně.
- c. *Odebrání knihy z knihovny:* Systém umožňuje přihlášenému uživateli odstranit vybranou knihu z jeho knihovny.
- d. *Vytvoření štítku:* Systém umožňuje uživateli vytvořit si štítky na filtraci knih. Štítek musí mít jméno a může mít popis.
- e. *Úprava štítku:* Systém umožňuje uživateli upravit popis štítku.
- f. *Odstranění štítku:* Systém umožňuje uživateli odstranit štítek, čímž se odstraní i z přidělených knih.
- g. *Přidání štítku ke knize:* Systém umožňuje přidat ke knize štítek.
- h. *Odebrání štítku z knihy:* Systém umožňuje odebrat štítek z knihy.
- i. *Prohlížení vlastní knihovny:* Systém umožňuje přihlášenému uživateli zobrazit knihy ze své knihovny. Kniha je možné filtrovat a řadit. Filtrace je možná pomocí štítků, či textově pomocí jména, autora nebo jiného textu. Řazení je možné pomocí autora, jména, data přidání, data dočtení, hodnocení a dalších.
- j. *Prohlížení knihovny konkrétního přítele:* Systém umožňuje prohlížení knihoven uživatelů, které má přihlášený uživatel v seznamu přátel.

3. Hodnocení knih

- a. *Přidání textového hodnocení:* Systém umožňuje přihlášenému uživateli přidat ke knize, kterou má v knihovně, textové hodnocení.
- b. *Úprava textového hodnocení:* Systém umožňuje přihlášenému uživateli upravit hodnocení, která přidal ke svým knihám.
- c. *Odebrání textového hodnocení:* Systém umožňuje přihlášenému uživateli odebrat hodnocení, která přidal.
- d. *Přidání číselného hodnocení:* Systém umožňuje přihlášenému uživateli přidat číselné hodnocení ke knize. Číselné hodnocení se přidává pomocí zvolení počtu hvězdiček mezi 1 a 5.
- e. *Úprava číselného hodnocení:* Systém umožňuje přihlášenému uživateli změnit číselné hodnocení přidané ke knize.
- f. *Odebrání číselného hodnocení:* Systém umožňuje přihlášenému uživateli odstranit číselné hodnocení, které knize dal.
- g. *Zobrazení vlastních hodnocení:* Systém umožňuje přihlášenému uživateli zobrazit seznam hodnocení, která sám ke knihám přidal. Uživateli je umožněno hodnocení řadit vzestupně nebo sestupně.

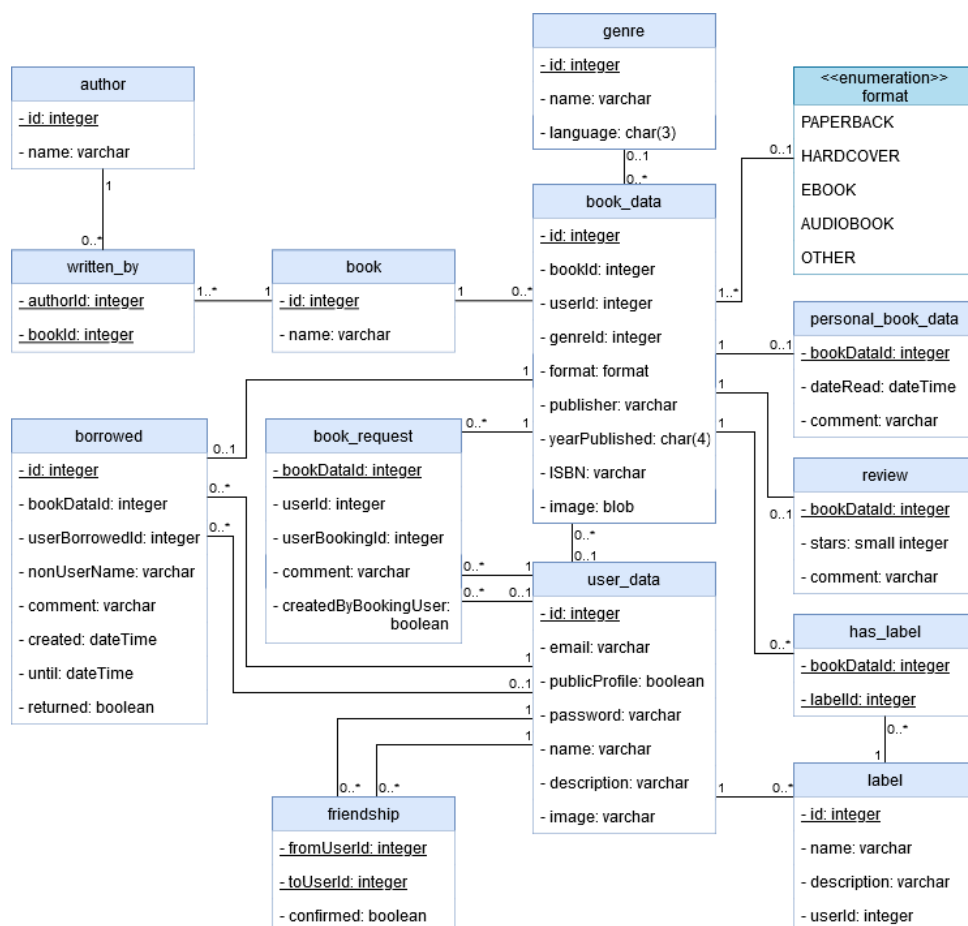
- b. *Upravení knihy ve svém wishlistu:* Systém umožňuje přihlášenému uživateli upravit si volitelné údaje o knize ve wishlistu.
- c. *Odebrání knihy ze svého wishlistu:* Systém umožňuje přihlášenému uživateli odebrat knihu, kterou si přidal do wishlistu.
- d. *Zobrazení svého wishlistu:* Systém umožňuje přihlášenému uživateli zobrazit si vlastní seznam přání, kde uvidí pouze knihy, které sám přidal.
- e. *Přidání knihy z wishlistu do knihovny:* Systém umožňuje přihlášenému uživateli přidat si knihu z wishlistu do knihovny. Při přidání může uživatel rovnou vyplnit všechny volitelné údaje či jí přidat štítky. Pokud přidá uživatel knihu, na kterou byla stopka, stopka je odstraněna.
- f. *Přidání knihy pořízené příteli mimo wishlist:* Systém umožňuje přihlášenému uživateli v rámci cizích seznamů přání přidat do nich knihu pořízenou tomuto uživateli. Přihlášený uživatel může také tuto knihu označit za darovanou, a tak jí přidat do knihovny obdarovaného. Jiní uživatelé ji budou mít zobrazenou pod wishlistem jako zabranou. U této knihy se zadávají údaje stejně jako při přidání do wishlistu. Toto přidání zapíná uživateli stopku, pokud není již zapnutá.
- g. *Editace knihy pořízené příteli mimo wishlist:* Systém umožňuje přihlášenému uživateli editovat údaje o knize, kterou zadal uživateli do pořízených mimo wishlist. Uživatel taky může přesunout knihu do wishlistu uživatele, čímž se připraví o možnost ji dále editovat.
- h. *Odstranění knihy pořízené příteli mimo wishlist:* Systém umožňuje přihlášenému uživateli odebrat knihu, kterou přidal příteli do seznamu knih, které mu chce darovat. Pokud to byla jediná kniha, kvůli které existovala stopka, je odstraněna.
- i. *Zobrazení wishlistu přátel:* Systém umožňuje přihlášenému uživateli zobrazit si wishlist vybraného přítele.
- j. *Zobrazení stopky na pořizování knih:* Systém umožňuje přihlášenému uživateli zobrazit si, zda pro něj má někdo pořízenou knihu jako dárek. Systém neumožňuje zobrazit, kdo pro něj má jakou knihu, jen že existuje nějaká zabraná kniha v jeho wishlistu či přidaná do pořízených pro něj.
- k. *Označení knihy ve wishlistu přítele jako zamluvené:* Systém umožňuje přihlášenému uživateli zmluvit si pořízení knihy ve wishlistu přítele. Tato akce zapíná uživateli stopku, pokud není již zapnutá.
- l. *Odoznačení zamluvené knihy ve wishlistu přítele:* Systém umožňuje přihlášenému uživateli odebrat zmluvení knihy z wishlistu.



Obrázek 5.2: Případy užití

5.3 Databázový model

Na základě doménového modelu a detailní specifikace systémových požadavků jsem vytvořila databázové schéma, viz obrázek 5.3, na kterém je aplikace postavena.



Obrázek 5.3: Diagram databázového modelu

5.4 API

Návrh serverové části jsem blíže specifikovala pomocí API¹, které implementované aplikace poskytuje.

Aplikace využívá principy REST² API, tedy umožňuje provádět jednoduchá CRUD³ volání nad aplikačními daty pomocí HTTP⁴.

¹Application Programming Interface

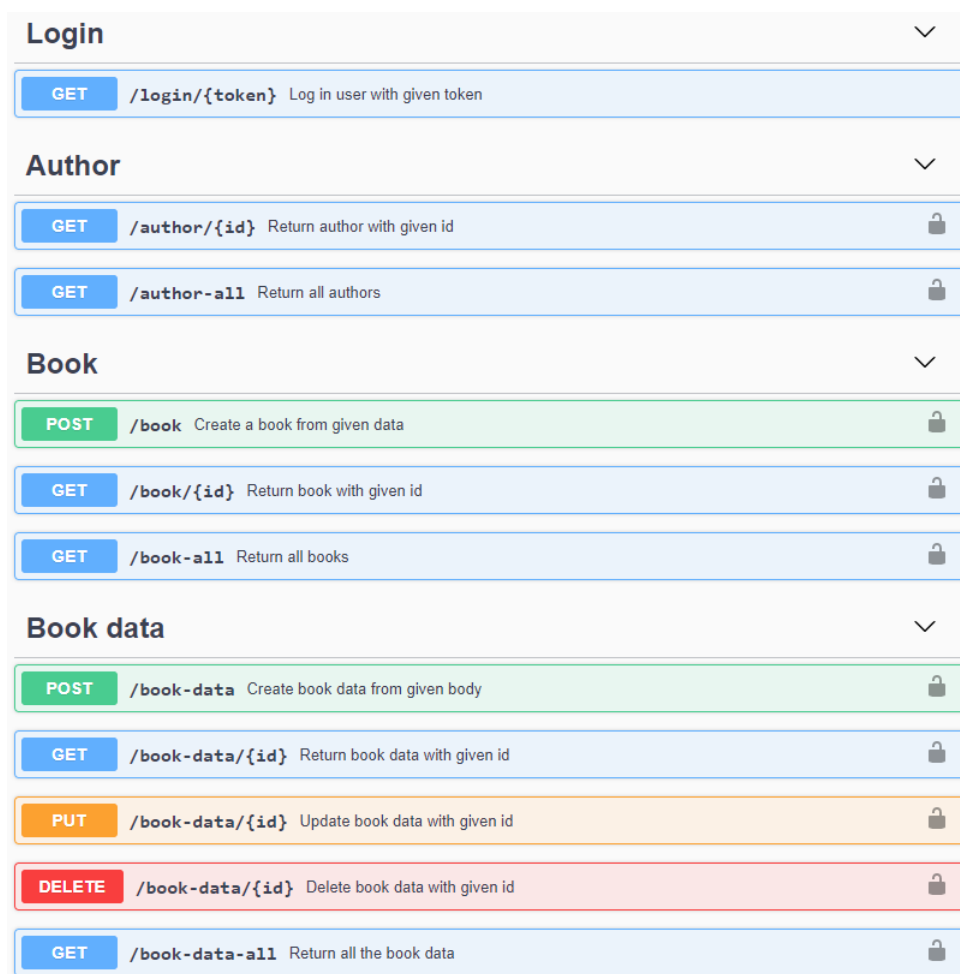
²Representational State Transfer

³Create, Read, Update, Delete

⁴Hypertext Transfer Protocol

Pro detailnější popis REST API implementované aplikace a dat, se kterými pracuje, jsem zvolila OpenAPI [8], dříve známé jako Swagger. OpenAPI funguje jako přehledná dokumentace a zároveň poskytuje možnost přímé interakce a testování dotazů. Díky schopnosti autorizace je v OpenAPI také možné jednoduše testovat volání zabezpečených dotazů.

Ve vytvořené dokumentaci jsem se soustředila převážně na jednoduché popsání dotazů, předávaných parametrů a návratových hodnot.



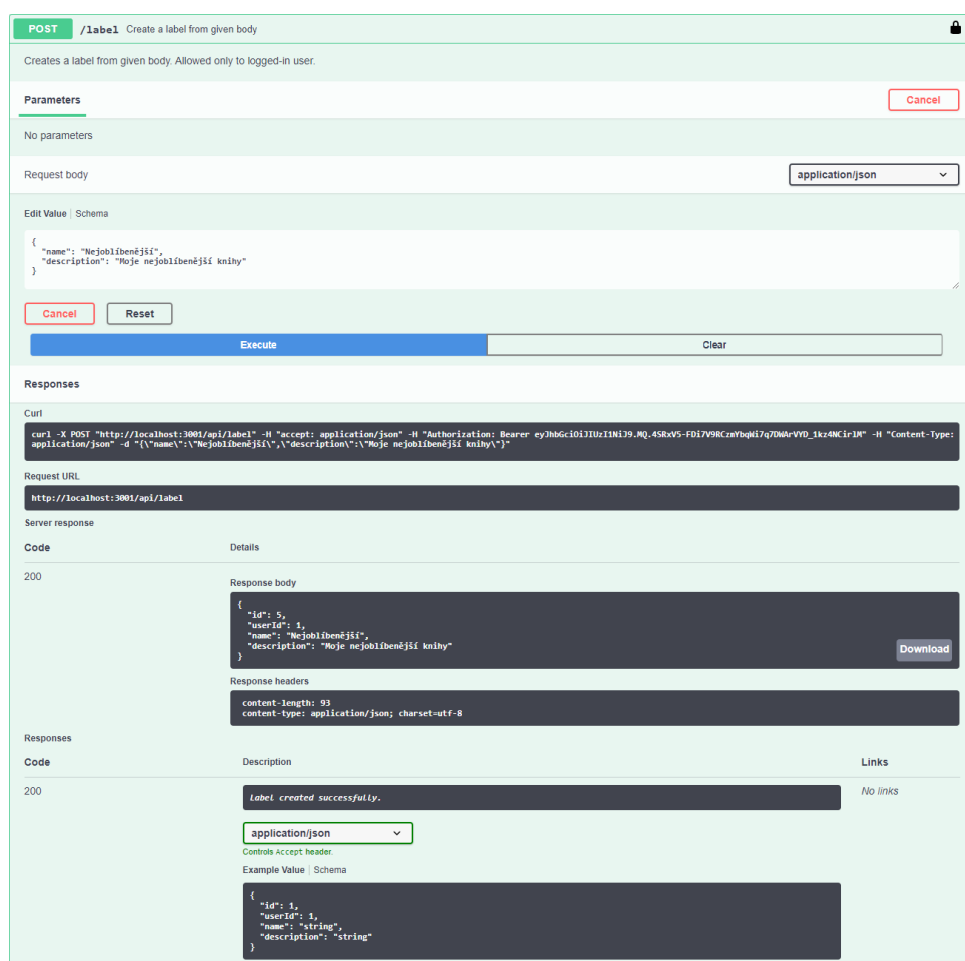
Obrázek 5.4: Ukázka dotazů OpenAPI

Na obrázku 5.4 je ukázka dotazů popsaných OpenAPI, na obrázku 5.5 je zobrazen detail jednoho takového dotazu po jeho zavolání.

Celá dokumentace se nachází v kořenovém adresáři serverové části aplikace.

5.5 Prototyp uživatelského rozhraní

Pro návrh klientské části jsem se rozhodla využít prototypování. Mým cílem bylo vytvoření několika statických obrazovek, které by mi napomohly v ucelení si obrazu o uživatelském rozhraní a budoucím vzhledu aplikace. Tyto



Obrázek 5.5: Detail dotazu OpenAPI

obrazovky jsem také chtěla konzultovat s potenciálními uživateli aplikace a případně je podle jejich připomínek přizpůsobit.

Po zvážení mých požadavků a současných zkušeností jsem vybrala k vytvoření tohoto prototypu nástroj Axure [9]. Axure pracuje s prototypy v podobě HTML⁵ [10] a tak je vhodný na prototypování UI⁶ webového klienta. Také umožňuje přizpůsobení míry komplexity dle požadavků a jednoduché sdílení za použití odkazu [11]. K tomuto výběru také přispěl fakt, že v současnosti poskytuje studentskou licenci zdarma.

V rámci prototypování jsem tedy vytvořila několik obrazovek, z čehož první byla obrazovka pro přihlášení a registraci, neboť je nutná pro začátek práce s aplikací. Pro tuto obrazovku již existuje mnoho hojně využívaných standardů, na které jsou uživatelé webových aplikací zvyklí, a je tak se ukázalo nejvhodnější jeden z nich zvolit a přizpůsobit své aplikaci. Obrazovka je na obrázku 5.6.

Následně jsem navrhla obrazovku pro zobrazování knihovny. Hlavní motivací

⁵Hypertext Markup Language

⁶User Interface – uživatelské rozhraní

Přihlášení

Email

Heslo

☐ Neodhlašovat

[zapomněli jste heslo?](#)

pomoci Google pomoci Facebooku

Registrace

Email*

Heslo*

☐ Souhlasím s [podmínkami užítí](#)

pomoci Google pomoci Facebooku

O aplikaci

Vítejte na Knihappce, aplikaci určené pro všechny čtenáře knih.
 V aplikaci si lze udržovat seznamy svých knih a sdílet jejich recenze (veřejně či mezi přáteli).
 Aplikace usnadňuje udržování seznamů zapůjčených knih a také umožňuje přátelům zobrazit si Váš wishlist knih a označit v něm knihu, kterou vám koupí. Ostatní přátelé tak uvidí, jakou knihu již nemají kupovat a Vy uvidíte stopku na kupování knih.

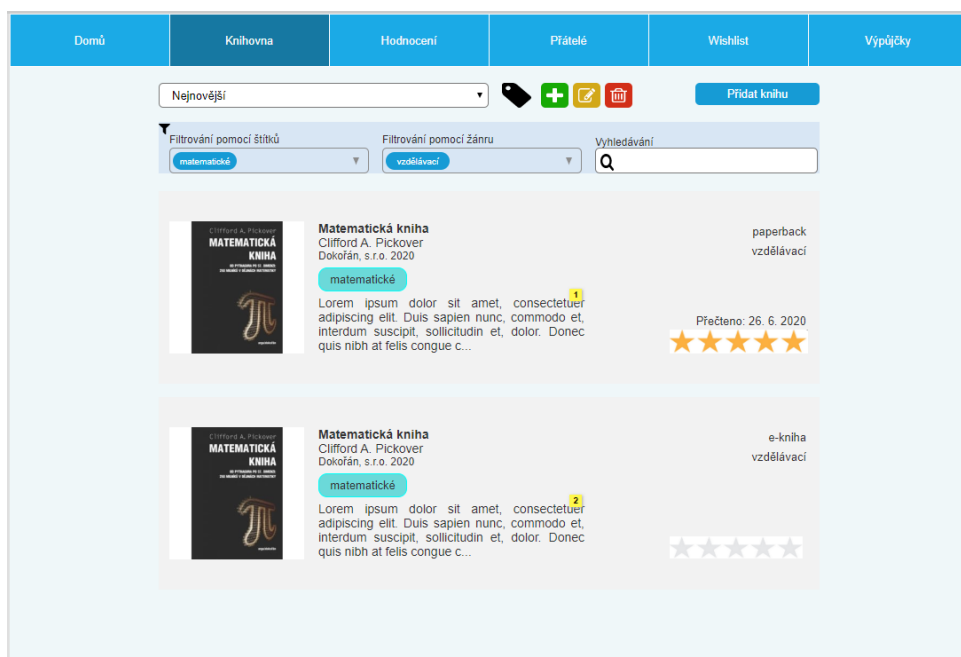
Obrázek 5.6: Prototyp přihlášení a registrace

bylo probrat s uživateli, jaká data je vhodná zobrazit v kartě knihy a jaké množství dat je ještě přehledné. Záměrem nebylo zobrazit všechna data, ta jsou zobrazena až v detailu knihy, pro přehlednost rozdělená do kategorií. Poslední verzi upravenou dle konzultací s uživateli lze vidět na obrázku 5.7.

Dále jsem navrhla několik obrazovek, abych zjistila, zda jsou pro jednoduché operace výhodnější samostatné obrazovky či vyskakovací okna. Díky konzultacím s potenciálními uživateli v této fázi jsem došla k závěru, že původní návrh vyskakovacích oken nepoužiji. Hlavním důvodem bylo to, že uživateli je při použití vyskakovacího okna zabráněna interakce s jinými částmi aplikace a akce tak musí být vždy vykonána nebo stornována. Nelze si tak například otevřít detail knihy v jiném panelu. Vyskakovací okna se však do finální aplikace dostala na potvrzení operací, které nevratně mažou data, jako například odstranění knihy nebo profilu.

V návrhu jsem také vytvořila obrazovku zadávání nové knihy pro upřesnění způsobu vkládání konkrétních typů dat.

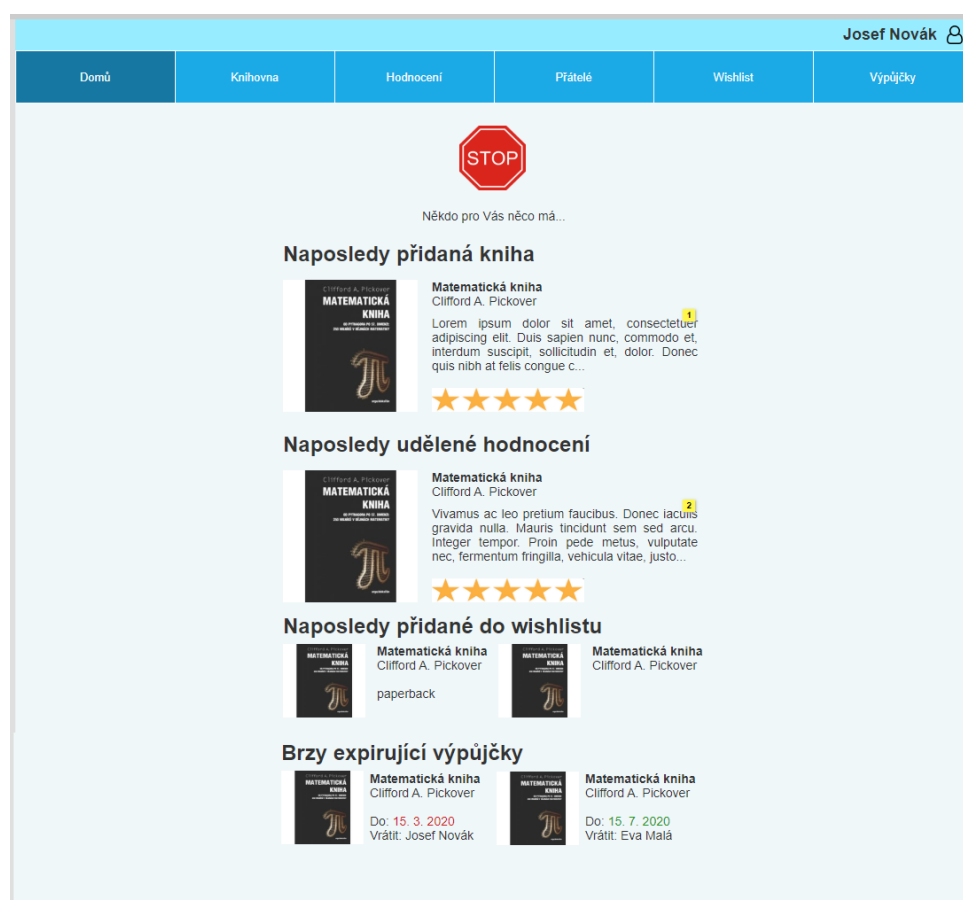
Jako poslední obrazovku jsem vytvořila domovskou stránku, tedy stránku, která se zobrazí každému uživateli po přihlášení do aplikace. Zde byly konzultace s uživateli nejdůležitější, neboť šlo hlavně o snahu poskytnout jim to nejrelevantnější z celé aplikace na malém prostoru. Výsledek těchto konzultací je zobrazen na obrázku 5.8.



Obrázek 5.7: Prototyp knihovny

5.6 Shrnutí kapitoly

Nejdříve jsem popsala doménový model implementované aplikace, který je založený na požadavcích z kapitoly Funkční požadavky. Pokračovala jsem popisem systémových požadavků, které jsem graficky znázornila pomocí případů užití. Systémové požadavky jsem popsala pomocí bližší specifikace uživatelských požadavků. Navázala jsem databázovým modelem, který rozšiřuje doménový model a upravuje jej tak, aby byl vhodný pro sestavení databáze aplikace. Pro bližší popis serverové části aplikace jsem sepsala specifikaci REST API, pro vizualizaci klientské části jsem následně vytvořila prototyp aplikace.



Obrázek 5.8: Prototyp domovské stránky

Kapitola 6

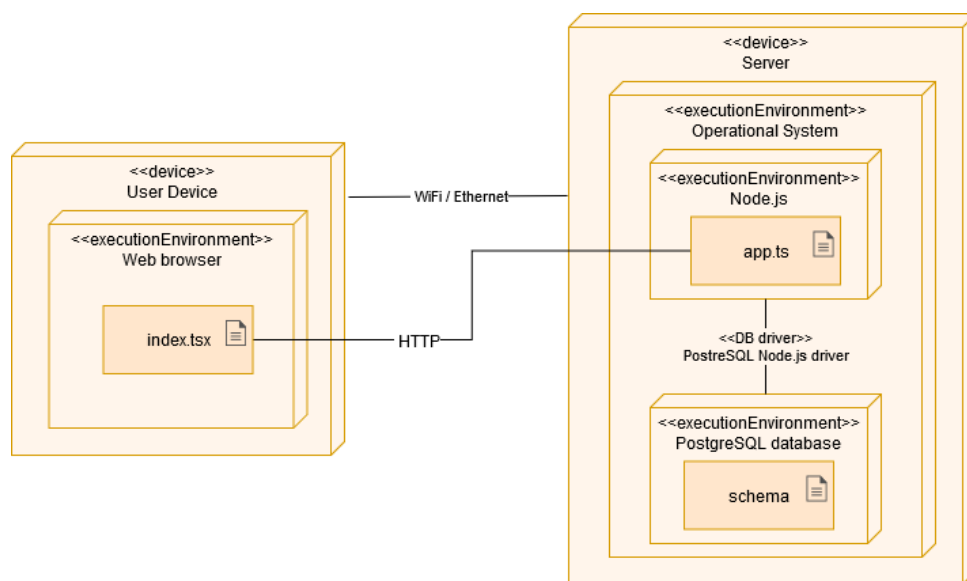
Implementace a nasazení aplikace

V této kapitole popisuji, jaké technologie jsem zvolila k implementaci aplikace a jak funguje její sestavení a instalace.

6.1 Architektura aplikace

Implementovaná aplikace se skládá z databáze, klientské části a serverové části.

Diagram nasazení jednotlivých částí aplikace je zobrazen na obrázku 6.1.



Obrázek 6.1: Nasazení aplikace

6.1.1 Databáze

Při výběru vhodné databáze jsem zvažovala možnost SQL¹ i NoSQL² standardu. Zvolila jsem relační databázi standardu SQL, neboť varianty NoSQL

¹Structured Query Language

²Not only SQL

K napojení na REST API v rámci serverové části jsem zvolila knihovnu Axios [23], která je standardně využívána k tomuto účelu.

6.1.4 Použití TypeScriptu

Hlavní nevýhodou jazyka JavaScript, použitého ve zmíněných technologiích, je absence statického typování. Nedochází tedy ke kompilaci, při níž jsou ověřeny typy jednotlivých proměnných, parametrů a návratových hodnot. Mnoho chyb se tak projeví až za běhu aplikace a nemusí být ani zřejmé, zejména kvůli automatickým typovým konverzím, které jazyk dynamicky provádí. Tento fakt komplikuje údržbu aplikací většího rozsahu a případné odhalování chyb.

Zmíněnou nevýhodu se snaží eliminovat jazyková nadstavba TypeScript [24] přidávající k JavaScriptu zejména statickou kontrolu typů. Typový systém jazyka je založen na principu strukturálního typování, v němž si dva typy odpovídají, pokud je kompatibilní jejich struktura. Díky tomu je možné přiřadit typy i k již existujícím funkcím napsaným v JavaScriptu.

I přes zpomalení vývoje z důvodu nutného vytváření typů jsem zvolila TypeScript kvůli tomu, že aplikace je zaměřena jako open-source, a použití TypeScriptu značně zvyšuje čitelnost kódu a usnadňuje jeho případné úpravy. Typy samotné fungují jako dokumentace a usnadňují správné použití jednotlivých funkcí.

6.1.5 Sdílení kódu mezi klientskou a serverovou částí

Pro fungování aplikace je nutné, aby serverová a klientská část komunikovaly prostřednictvím zasílání zpráv s daty. Jako formát, ve kterém se budou tato data posílat, jsem kvůli jejich struktuře a použitým technologiím zvolila standard JSON⁵. Zajistila jsem, aby tato data interpretovala stejným způsobem klientská i serverová část aplikace, což bylo díky použití stejného jazyka pro obě části možné pomocí sdílení kódu obsahujícího typy těchto objektů. Sdílení jsem využila i pro funkcionality sloužící k jejich verifikaci a zpracování.

Nejjednodušším způsobem pro toto sdílení je duplikace kódu mezi klientskou a serverovou částí aplikace. V takovém případě je nutné při každé změně upravovat obě verze tohoto duplikátu. Kód tím ztratí princip SSOT⁶, který zajišťuje konzistenci dat a přístupu k nim.

Výhodnější je tedy mít sdílené typy a funkce pouze v jednom adresáři. React však v současnosti za použití základního nastavení neumožňuje využití TypeScriptového kódu mimo vyhrazené adresáře uvnitř jeho struktury. Proto by bylo nutné, aby se tento adresář nacházel uvnitř klientské části projektu. To je sice možné díky tomu, že Node.js žádná takováto omezení nemá, ovšem dojde k narušení principu SOC⁷, neboť jsou data potřebná k běhu serveru obsažena v klientské části, a řešení se tak může jevit jako značně matoucí.

⁵JavaScript Object Notation

⁶Single Source Of Truth

⁷Separation Of Concerns

6.2.1 Potřebné technologie

K instalaci aplikace a jejímu vývoji je třeba mít na zařízení nainstalovaný PostgreSQL [27], spolu s vhodným programem na jeho ovládání, a Node.js [28].

PostgreSQL slouží k běhu databáze. K jejímu vytvoření je vhodné použít některý z programů usnadňujících práci s databázemi, například pgAdmin [29] či DataGrip [30]. Oba dva tyto programy byly použity i v rámci implementace této práce.

Node.js pak slouží jak ke správnému běhu serverové části, tak ke správě knihoven pomocí npm¹¹. Aplikace byla vyvíjena na verzi 13.9.0, ale testována byla až na verzi 8.16.0 bez výrazných odlišností. O jakékoli další instalace se stará přímo npm.

6.2.2 Popis instalace

Uvnitř staženého repozitáře je potřeba provést následující kroky:

1. *Vytvoření databáze:* Vytvořte prázdnou PostgreSQL databázi pro aplikaci.
2. *Vytvoření databázových tabulek:* Spustěte v této databázi všechny skripty obsažené v adresáři s migracemi (`/packages/backend/migrations`) v daném pořadí.
3. *Povolení Google autorizace:* Použijte vaše Google OAuth2 klientské id [31]. Pokud na něm nemáte povolený přístup a přesměrování na adresu `localhost:3000`, je třeba změnit tato povolení v nastavení. Pokud toto id nemáte, vytvořte si jej v Google API Console [32] a potřebná povolení mu nastavte.
4. *Určení proměnných prostředí:* Vytvořte `.env` soubor v kořenovém adresáři serveru (`/packages/backend/`). Do tohoto souboru přidejte hodnoty dle popisu umístěném zde v `.env-description.md`. To zopakujte pro kořenový adresář klientské aplikace (`/packages/frontend/`), kde se nachází `.env-description.md`.
5. *Spuštění instalace:* Pro spuštění instalace spustěte v kořenovém adresáři celého projektu příkaz `npm run bootstrap` pro nainstalování všech závislostí.

6.2.3 Popis spuštění aplikace

Spuštění aplikace je možné pomocí příkazu `npm run start-dev` v kořenovém adresáři projektu. Spuštění produkčního režimu není v alfa verzi doporučováno.

¹¹Node.js package manager

fungování zabezpečení aplikace.

Na závěr jsem popsala instalaci a sestavení aplikace, vyjmenovala potřebné technologie a sepsala návod k instalaci. Také jsem uvedla, jak takto nainstalovanou aplikaci spustit.

Kapitola 7

Testování aplikace

Tato kapitola se zaměřuje na specifikaci uživatelských testů a zaznamenání a vyhodnocení jejich výsledků.

7.1 Podmínky testování

Aplikace byla testována během vývoje, dříve, než byla nasazena na server. Testování uživatelé tak museli mít přístup k zařízení, na kterém byl sestavený projekt, nebo jej museli být schopni sami sestavit. První možnost byla značně redukována současnou situací (omezení osobních setkávání kvůli regulacím souvisejícím s COVID-19), pro druhou možnost bylo nutné se zaměřit na dostatečně technicky zdatné uživatele. Částečnou výhodou druhé možnosti bylo i otestování, zda technicky zdatní uživatelé budou schopni sestavit aplikaci podle návodu na sestavení. Pro získání dostatečného vzorku byly nakonec testy prováděny s některými lidmi osobně a s ostatními za pomoci chatu a videohovorů pomocí služeb Messenger a Slack.

7.2 Cílová skupina

Cílovou skupinou tohoto testování jsou čtenáři, vlastníci knih a jejich přátelé, kteří by chtěli využívat možnosti aplikace. Testování byli lidé s různou úrovní schopností práce s počítačem a webovými aplikacemi, od těch schopných si aplikaci sestavit, po občasné uživatele informačních technologií. Účastníci testu souhlasili také se sdílením své věku pro případnou relevantnost této informace.

7.3 Testovací scénáře

Kvůli kvantitativnímu omezení jsem testy zaměřila na detailnější průchod aplikací. Navrhla jsem několik testovacích scénářů s minimální sadou instrukcí a zkoumala jsem, zda se testované osoby budou schopny snadno orientovat a nenarazí na žádnou chybu nebo omezení. Během testů byly také sepsány jejich případné další návrhy mimo scénáře.

Testovací scénáře byly tyto:

1. *Registrujte se:* Registrujte se do aplikace, uveďte své jméno, aby vás přátelé jednodušeji poznali.
2. *Odhlaste se:* Odhlaste se z aplikace.
3. *Přidejte si uživatele do přátel:* Registrujte si jiný účet a přidejte si Váš první účet mezi přátele.
4. *Přidejte si knihu do knihovny:* Přidejte si v aplikaci do své knihovny některou ze svých knih. Vyplňte všechny údaje.
5. *Upravte datum přečtení této knihy:* Upravte pro právě přidanou knihu datum, kdy jste ji přečetli.
6. *Vytvořte si štítek:* Vytvořte štítek a přidejte jej k přidané knize.
7. *Zapůjčte příteli knihu:* Vytvořte záznam o půjčení vaší knihy příteli.
8. *Označte knihu jako vrácenou:* Označte zapůjčenou knihu za vrácenou.
9. *Přidejte si knihu do wishlist:* Přidejte si do wishlistu knihu, kterou si přejete, a zadejte k ní dostatečné údaje.
10. *Zamluvte si knihu ve wishlistu přítele:* Vyberte si knihu z wishlistu přítele a označte ji jako zamluvenou.
11. *Odstraňte si knížku z knihovny:* Odstraňte si knihu, kterou jste dříve přidali do knihovny.
12. *Smažte si účet:* Odstraňte si účet v aplikaci.

7.4 Výsledky testování

Většina testujících uživatelů neměla výrazný problém splnit všechny testovací scénáře, proto jsem se zaměřila více na sběr další zpětné vazby a objevené chyby a nejasnosti. Testujícím bylo oznámeno, že aplikace je v alfa verzi a byly jim stručně shrnuta plánovaná rozšíření. Po každém testu byly ihned provedeny opravy, aby další testující měl k dispozici verzi se zapracovanými připomínkami.

Prvním testujícím uživatelem byl muž (33 let), programátor, který není rodilý mluvčí.

- Testovací scénáře prošel bez problémů.
- Ve scénáři 8 testující očekával, že tlačítko pro vrácení bude spíše na stránce přehledu než v detailu půjčky. Připomínku jsem zapracovala.
- Testující zkonstatoval, že v detailu by očekával místo tlačítka odstranění tlačítko zpět. Odstranění by znovu čekal spíše v přehledu. Připomínku jsem zapracovala.

Druhým testujícím byl muž (30 let), specialista na IT bezpečnost.

- Testovací scénáře prošel bez problémů.
- V průběhu scénáře 4 testující zjistil, že test validity ISBN neprochází i na validní data. Nalezená chyba byla opravena.
- Testující během scénáře 7 nepochopil, k čemu slouží možnost zapsat si k výpůjčce i jméno uživatele. K poli jsem pro upřesnění přidala popis.
- Testujícímu by přišlo vhodnější, kdyby tlačítko zpět ve formulářích vracelo spíše na obrazovku sekce než na obrazovku, ze které se tam dostal. Vzhledem k rozdílným názorům na požadovanou funkcionalitu zpět tlačítka nebyla tato připomínka zapracována.

Třetím testujícím byl muž (28 let), novinář

- Testovací scénáře prošel téměř bez problémů, zasekl se pouze na pokusu editovat email na registraci, který slouží pouze ke čtení. Pole pro čtení jsem upravila, aby bylo více jasné k čemu slouží.
- Testující upozornil, že zprávy při selhání registrace a přihlášení mají nejasnou formu. Zapracovala jsem možnost upravit zprávu na konkrétní selhání a tyto dvě zprávy jsem ujasnila.
- Testující navrhl, že jednotlivá podmenu by mohla být přehledněji přepínatelná. Vylepšení designu je plánované po konzultaci s UX¹ designerem mimo rozsah této práce.

Čtvrtým testujícím byl muž (24 let), student IT

- Testovací scénáře prošel bez problémů, ale vynechal body 3 a 10, kvůli lokální nedostupnosti druhého Google účtu.
- Testující navrhl, že uložení editace by mělo vracet na stránku, ze které editaci spustil, tedy detail nebo přehled. Tuto připomínku jsem zapracovala.
- Testující ve scénáři 8 navrhl změnu popisu tlačítka z 'vrátit' na 'vráceno', aby bylo jasnější, že uživatel označuje, že kniha byla vrácena jemu. Poznámku jsem zapracovala a pokusila se lépe oddělit půjčené knihy a knihy, které uživatel zapůjčil. Bohužel jsem zde narazila na omezení češtiny, která nedokáže rozdílně popsat jednoznačně jedním slovem, tak jsem alespoň rozvedla nadpisy stran.
- Testující si nebyl jistý, zda jsou nutná upozornění po úspěšně přidání akcí. Jiní uživatelé s tímto naopak byli spokojeni, takže jsem do plánu budoucích rozšíření přidala možnost nastavení vypnutí těchto upozornění.

Pátým testujícím byl muž (28 let), manažer

¹User Experience

- Testovací scénáře prošel bez problémů.
- Testující při plnění scénáře 1 zmínil, že při zavření Google okna by aplikace neměla ukazovat chybu. Tento návrh byl zapracován.
- Testující navrhl, že pro zadávání popisu by se mělo pole rozšiřovat na více řádek, aby si ho po sobě mohl přečíst i když přesahuje rozsah řádky. Připomínka byla také zapracována.
- Testující ve scénáři 4 poznamenal, že kniha by měla mít možnost přidání více autorů. Tato funkcionality je v aplikaci plánována, chybí přidání formulářového pole, které toto umožňuje.

Šestáým testujícím byl muž (28 let), programátor

- Testovací scénáře prošel bez problémů.
- Testující si všiml, že po uložení dat stránka na chvíli zobrazí stránku, na kterou přesměrovává s původními daty, až poté zobrazí načítání a pak nová data. Nehodnotil to ale jako velký problém a tato chyba tak bude opravena až pro další verzi aplikace.
- Testující navrhl přidání několika nápověd do aplikace, jako například určení, kdo uvidí poznámku k zapůjčení knihy. Tyto nápovědy byly přidány do aplikace.
- Testující poznamenal, že by očekával, že po kliknutí na logo bude přesměrován na hlavní stránku aplikace. Protože se jedná o vcelku běžnou funkcionality, byla do aplikace také přidána.

7.5 Závěry testování

Aplikaci dle scénářů testovalo 6 uživatelů. Jednalo se o muže v rozmezí 24-30 let. Testovací vzorek tak byl značně omezen a plánuji jej pro další testování aplikace rozšířit. Další testování je plánované po nasazení beta verze mimo rozsah této práce.

Všichni testující byli schopni samostatně dokončit testovací scénáře. Testující našli několik chyb a méně přehledných míst, které jsem upravila dle diskuze s nimi. Podali také několik dalších návrhů, které jsem přidala do plánu budoucích rozšíření aplikace.

Aplikace z testů vyšla jako jednoduše použitelná a přehledná. Výsledná verze, po provedení navrhovaných oprav, má nedostatky zvláště v designu a možném lépe vypadajícím uživatelském rozhraním. Z funkcionalit, které jsou plánovány pro další verzi, by testující oceňovali nejvíce možnost filtrace, řazení a stránkování.

■ 7.6 Shrnutí kapitoly

V této kapitole jsem popsala podmínky, za kterých probíhalo testování aplikace, rozebrala skupinu, na kterou bylo testování zacíleno a rozepsala testovací scénáře, které byly k tomuto testování použity. Dále jsem vylicila jednotlivé průběhy testů a popsala k jakým změnám v aplikaci vedly. Nakonec jsem vyhodnotila závěry, které z testování vyplynuly.

Kapitola 8

Stav aplikace

Tato kapitola se zabývá současným stavem aplikace, rozbořem aktuálních nedostatků a popisem budoucích rozšíření.

8.1 Současný stav

Aplikace se nachází ve verzi alfa, nejsou tak plně podporovány všechny navržené funkcionality. Také jsem během vývoje a testování narazila na další možná vylepšení aplikace, která nebyla z časových důvodů zapracována, ale budou součástí další verze. Po vyřešení nedostatků bude aplikace nasazena v beta verzi.

Aplikace je v době odevzdání práce nasazena na PaaS¹ Heroku [33]. Aplikace zde dostupná [34] je aktualizována automaticky při každé změně větve `develop` provedené v GitHub repozitáři.

8.1.1 Nedostatky serverové části

Hlavním nedostatkem serverové části aplikace je, že současná verze nepodporuje jinou formu přihlášení než pomocí účtu Google. Plánované je přidání autorizace i pomocí Facebook účtu a do budoucna i pomocí registrace v aplikaci, která bude potřebovat vlastní zabezpečení. Také k ní bude potřeba implementace ochrany proti zapomenutí hesla.

Menším nedostatkem je, že v současnosti není vyřešena validace ukládání obrázků do databáze. Navíc jsem v průběhu vývoje narazila i na jiné možnosti ukládání obrázků, které se mohou jevit jako vhodnější a bylo by proto lepší věnovat se další rešerši a implementovat nejvhodnější řešení. Z těchto důvodů ani klientská část zatím obrázky nepracovává.

8.1.2 Nedostatky klientské části

Klientská část má více nedostatků než serverová, hlavně z důvodu složitosti uživatelského testování a nutnosti reakce na nepředvídané uživatelské vstupy. Klientská aplikace tak v současnosti nevyužívá všechny schopnosti serveru.

¹Platform as a Service

Kapitola 9

Závěr

Cílem této práce bylo provedení analýzy požadavků sledované komunity čtenářů a provedení rešerše řešení naplňujících do jisté míry tyto požadavky. Dále pak navržení, implementace alfa verze a uživatelské testování aplikace, která bude splňovat dané požadavky včetně nalezených nedostatků současných řešení.

V rámci této práce jsem prošla celým procesem vytvoření aplikace, od sběru požadavků až po uživatelské testování aplikace, a zachytila v ní všechny provedené kroky. Z časových důvodů nebyly implementovány všechny požadavky sledované komunity, ale splněných bylo nad 90 %, konkrétně u případů užití 40 ze 44. To, vzhledem k tomu, že aplikace byla implementována v alfa verzi, hodnotím jako úspěšné. Dle mého názoru jsem zadání splnila.

V práci jsem popsala nejdříve teoretické základy lidské komunikace a rozdělení komunikačních kanálů a sociálních skupin, což mi pomohlo v lepším porozumění požadavků cílové skupiny a její specifikaci. Dále jsem se věnovala specifikaci obecných funkčních a nefunkčních požadavků, které by měla aplikace, kterou vyžaduje cílová skupina, splňovat. V další části jsem analyzovala v této komunitě často používané aplikace Goodreads a Databázi knih a zjišťovala jejich silné stránky i nedostatky. Pokračovala jsem návrhem aplikace, kde jsem postupně popsala doménový model, případy užití, systémové požadavky, databázový model, API a zkonstruovala návrh několika obrazovek. Po finalizaci návrhu jsem se zabývala architekturou aplikace, výběrem technologií a následovala implementací aplikace. K této implementaci jsem popsala potřebné technologie, návod její instalace a spuštění. Následně jsem navrhla scénáře uživatelských testů, implementovanou alfa verzi aplikace dala otestovat několika čtenářům a sepsala a zapracovala výsledky těchto testů. Nakonec jsem popsala stav aplikace v době jejího odevzdání, zmínila její nedostatky proti návrhu a uvedla budoucí rozšíření, na které jsem přišla během vývoje a testování.

Dalším krokem ve vývoji aplikace je odstranění nedostatků popsaných v kapitole Stav aplikace sekci Současný stav. Následně aplikaci vydám v beta verzi. Tato verze poté projde širším uživatelským testováním. Na další rozvoj aplikace využiji výstupy těchto testů a návrhy zaznamenané v sekci Možná rozšíření

Výsledná aplikace může sloužit jako zajímavé rozšíření portfolia existující-

cích aplikací zaznamenávajících knihovnu uživatele. Aplikace cílí zvláště na uživatele, kteří by ji rádi využívali pro udržování seznamů půjčených knih nebo wishlistů.

Pro mě byl největší přínos této práce právě průchod celým procesem tvorby aplikace od požadavků až po první nasazenou verzi. Vyjma procesu samotného mi práce značně pomohla i s pochopením mnoha teoretických pojmů či smyslů různých v praxi používaných diagramů. V neposlední řadě mi implementace aplikace pomohla rozvinout své programátorské znalosti a rozšířila mi přehled o různých moderních technologiích.



Literatura

- [1] Cambridge Dictionary – Information exchange. Dostupné na:
<https://dictionary.cambridge.org/us/dictionary/english/information-exchange>, 17. 12. 2019
- [2] Studium Psychologie – skupiny. Dostupné na:
<https://www.studium-psychologie.cz/socialni-psychologie/6-skupiny.html>, 17. 12. 2019
- [3] Goodreads. Dostupné na:
<https://www.goodreads.com/>, 11. 12. 2019
- [4] O Goodreads. Dostupné na:
<https://www.goodreads.com/about/us>, 10. 12. 2019
- [5] Databáze knih. Dostupné na:
<https://www.databazeknih.cz/>, 17. 12. 2019
- [6] Databáze knih – O nás. Dostupné na:
<https://www.databazeknih.cz/o-nas>, 17. 12. 2019
- [7] Česko-Slovenská filmová databáze. Dostupná na:
<https://www.csfd.cz/>, 5. 4. 2020
- [8] Specifikace OpenAPI. Dostupné na:
<https://github.com/OAI/OpenAPI-Specification>, 5. 4. 2020
- [9] Dokumentace k prototypujícímu nástroji Axure. Dostupné na:
<https://www.axure.com/>, 5. 4. 2020
- [10] HTML reference na MDN. Dostupné na:
<https://developer.mozilla.org/en-US/docs/Web/HTML>, 11. 5. 2020
- [11] Axure – odkaz na prototyp. Dostupné na:
<https://h4zgsu.axshare.com>, 11. 5. 2020

- [12] MongoDB dokumentace. Dostupná na:
<https://docs.mongodb.com/manual/>, 11. 5. 2020
- [13] GraphQL dokumentace. Dostupná na:
<https://graphql.org/learn/>, 11. 5. 2020
- [14] PostgreSQL dokumentace. Dostupná na:
<https://www.postgresql.org/docs/>, 11. 5. 2020
- [15] Databázové trendy 2019. Dostupné na:
<https://scalegrid.io/blog/2019-database-trends-sql-vs-nosql-top-databases-single-vs-multiple-database-use/>, 8. 1. 2020
- [16] Dokumentace Node.js. Dostupná na:
<https://nodejs.org/en/docs/>, 5. 4. 2020
- [17] Dokumentace Node.js knihovny Express. Dostupná na:
<https://expressjs.com/>, 12. 5. 2020
- [18] Dokumentace knihovny React. Dostupná na:
<https://reactjs.org/docs/>, 5. 4. 2020
- [19] Dokumentace knihovny Redux. Dostupná na:
<https://redux.js.org/>, 11. 5. 2020
- [20] Dokumentace knihovny Redux-Saga. Dostupná na:
<https://redux-saga.js.org/>, 11. 5. 2020
- [21] Dokumentace React knihovny Material-UI. Dostupná na:
<https://material-ui.com/getting-started/usage/>, 11. 5. 2020
- [22] Dokumentace principu Material Design. Dostupná na:
<https://material.io/>, 11. 5. 2020
- [23] Dokumentace knihovny Axios. Dostupná na:
<https://github.com/axios/axios>, 11. 5. 2020
- [24] Dokumentace jazyka TypeScript. Dostupná na:
<https://www.typescriptlang.org/docs/home>, 5. 4. 2020
- [25] Dokumentace nástroje Lerna. Dostupná na:
<https://lerna.js.org/>, 11. 5. 2020
- [26] Aktualizovaná verze aplikace. Dostupná na:
<https://github.com/stetizu1/book-readers-app>, 12. 5. 2020

- [27] Instalace PostgreSQL. Dostupná na:
<https://www.postgresql.org/download/>, 12. 5. 2020
- [28] Instalace Node.js. Dostupná na:
<https://nodejs.org/en/download/>, 12. 5. 2020
- [29] Dokumentace a odkaz k instalaci pgAdmin. Dostupné na:
<https://www.pgadmin.org/>, 12. 5. 2020
- [30] Dokumentace a odkaz k instalaci DataGrip. Dostupné na:
<https://www.jetbrains.com/datagrip/>, 12. 5. 2020
- [31] Dokumentace Google OAuth id. Dostupná na:
<https://developers.google.com/identity/sign-in/web>, 12. 5. 2020
- [32] Google API Console. Dostupné na:
<https://console.developers.google.com/apis/credentials>, 12. 5. 2020
- [33] Domovská stránka PaaS Heroku. Dostupná na:
<https://www.heroku.com/>, 22. 5. 2020
- [34] Heroku adresa nasazené aplikace. Dostupné na:
<https://book-app-thesis.herokuapp.com/>, 22. 5. 2020
- [35] Domovská stránka Dribbble. Dostupná na: <https://dribbble.com/>,
22. 5. 2020